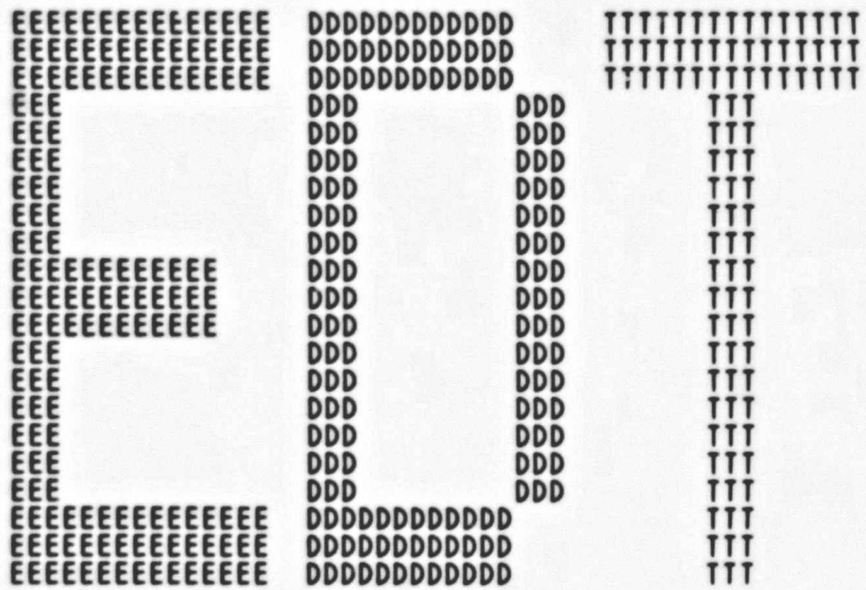
EEEEEEEEEEEEEEE DDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTT

FILEID**LXCOM

```
LL          XX      XX    CCCCCCCC    OOOOOO    MM      MM
LL          XX      XX    CCCCCCCC    OOOOOO    MM      MM
LL          XX      XX  CC            OO    OO  MMMM  MMMM
LL          XX      XX  CC            OO    OO  MMMM  MMMM
LL            XX  XX    CC            OO    OO  MM  MM  MM
LL            XX  XX    CC            OO    OO  MM  MM  MM
LL              XX      CC            OO    OO  MM      MM
LL              XX      CC            OO    OO  MM      MM
LL            XX  XX    CC            OO    OO  MM      MM
LL            XX  XX    CC            OO    OO  MM      MM
LL          XX      XX  CC            OO    OO  MM      MM      ....
LL          XX      XX  CC            OO    OO  MM      MM      ....
LLLLLLLLLL  XX      XX    CCCCCCCC    OOOOOO    MM      MM      ....
LLLLLLLLLL  XX      XX    CCCCCCCC    OOOOOO    MM      MM      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SSSSSS
LL              II      SSSSSS
LL              II            SS
LL              II            SS
LL              II            SS
LL              II            SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
;     1          0001  0 %TITLE 'EDT$LXCOM - execute a line-mode command'
;     2          0002  0 MODULE EDT$LXCOM (                              ! Execute a line-mode command
;     3          0003  0                   IDENT = 'V04-000'                     ! File: LXCOM.BLI Edit: JBS1020
;     4          0004  0                   ) =
;     5          0005  1 BEGIN
;     6          0006  1 !
;     7          0007  1 !**************************************************************************
;     8          0008  1 !*                                                                        *
;     9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                               *
;    10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
;    11          0011  1 !*  ALL RIGHTS RESERVED.                                                   *
;    12          0012  1 !*                                                                        *
;    13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    18          0018  1 !*  TRANSFERRED.                                                           *
;    19          0019  1 !*                                                                        *
;    20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    22          0022  1 !*  CORPORATION.                                                           *
;    23          0023  1 !*                                                                        *
;    24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
;    26          0026  1 !*                                                                        *
;    27          0027  1 !*                                                                        *
;    28          0028  1 !**************************************************************************
;    29          0029  1 !
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:     EDT -- The DEC Standard Editor
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       Execute a line-mode command.  Most of the work is done by external routines.
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT: User mode
;    39          0039  1 !
;    40          0040  1 ! AUTHOR: Bob Kushlis, CREATION DATE: February 3, 1978
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 ! 1-001 - Original.  DJS 10-MAR-81.  This module was created by
;    45          0045  1 !       extracting the routine EDT$$LNM_CMD  from EXEC.BLI.
;    46          0046  1 ! 1-002 - Fix module name.  JBS 23-Mar-1981
;    47          0047  1 ! 1-003 - Make sure the routine names start with EDT$$.  JBS 31-Mar-1981
;    48          0048  1 ! 1-004 - Use the ASSERT macro.  JBS 01-Jun-1981
;    49          0049  1 ! 1-005 - Use the new message codes.  JBS 04-Aug-1981
;    50          0050  1 ! 1-006 - Add a synonym for the entry point name, so the overlay analyzer
;    51          0051  1 !         can be told to ignore it, and make sure the EXTEND module
;    52          0052  1 !         is in memory before returning.  JBS 12-Mar-1982
;    53          0053  1 ! 1-007 - Revise the overlay structure, since the method outlined above
;    54          0054  1 !         can cause an undefined symbol.  JBS 15-Mar-1982
;    55          0055  1 ! 1-008 - Remove the reference to EDT$$LOAD_EXTEND.  JBS 18-Mar-1982
;    56          0056  1 ! 1-009 - Set stay flag if command is a write. STS 10-May-1982
;    57          0057  1 ! 1-010 - Bypass EDT$$LNM_HLP by calling EDT$$OUT_HLP directly.  JBS 03-Jun-1982
```

```
;    58          0058  1 ! 1-011 - Set EDT$$G_TXT_ONSCR for commands that don't.  SMB 11-Jun-1982
;    59          0059  1 ! 1-012 - Set stay flag if command is a print. STS 10-May-1982
;    60          0060  1 ! 1-013 - Modify the setting of message flags.  SMB 23-Jun-1982
;    61          0061  1 ! 1-014 - Use EDT$$FMT_CRLF instead of EDT$$OUT_FMTBUF.  This means we don't have
;    62          0062  1 !           to set EDT$$G_MSGFLG or EDT$$G_TXT_ONSCR in most cases.  JBS 05-Jul-1982
;    63          0063  1 ! 1-015 - Move setting of EDT$$G_SCR_CHGD to the routines called.  JBS 07-Jul-1982
;    64          0064  1 ! 1-016 - Set EDT$$G_SCR_CHG in FIND only if it succeeds.  JBS 07-Jul-1982
;    65          0065  1 ! 1-017 - Remove EDT$$G_SCR_CHGD, the new screen logic does not need it.  JBS 09-Oct-1982
;    66          0066  1 ! 1-018 - Put code for edt$$rng_posfrst in line. STS 11-Oct-1982
;    67          0067  1 ! 1-019 - Remove forced setting of G_TXT_ONSCR for insert and replace.  SMB 14-Dec-1982
;    68          0068  1 ! 1-020 - Correct a typo in a comment.  JBS 02-May-1983
;    69          0069  1 !--
;    70          0070  1
```

```
;    72          0071  1 %SBTTL 'Declarations'
;    73          0072  1 !
;    74          0073  1 ! TABLE OF CONTENTS:
;    75          0074  1 !
;    76          0075  1
;    77          0076  1 REQUIRE 'EDTSRC:TRAROUNAM';
;    78          0515  1
;    79          0516  1 FORWARD ROUTINE
;    80          0517  1     EDT$$LNM_CMD : NOVALUE;                     ! Execute the last parsed command
;    81          0518  1
;    82          0519  1 !
;    83          0520  1 ! INCLUDE FILES:
;    84          0521  1 !
;    85          0522  1
;    86          0523  1 REQUIRE 'EDTSRC:EDTREQ';
;    87          0658  1
;    88          0659  1 !
;    89          0660  1 ! MACROS:
;    90          0661  1 !
;    91          0662  1 !       NONE
;    92          0663  1 !
;    93          0664  1 ! EQUATED SYMBOLS:
;    94          0665  1 !
;    95          0666  1 !+
;    96          0667  1 ! The EXTEND change-mode command uses this symbol indirectly to refer
;    97          0668  1 ! to this module, so the reference can be excluded from
;    98          0669  1 ! the overlay analysis.
;    99          0670  1 !-
;   100          0671  1
;   101          0672  1 GLOBAL BIND
;   102          0673  1     ROUTINE
;   103          0674  1     EDT$$LNM_CMD_NOOVERLAY_REF = EDT$$LNM_CMD;
;   104          0675  1
;   105          0676  1 !
;   106          0677  1 ! OWN STORAGE:
;   107          0678  1 !
;   108          0679  1 !       NONE
;   109          0680  1 !
;   110          0681  1 ! EXTERNAL REFERENCES:
;   111          0682  1 !
;   112          0683  1 !       In the routines
```

```
;   114          0684  1 %SBTTL 'EDT$$LNM_CMD  - execute a line-mode command'
;   115          0685  1
;   116          0686  1 GLOBAL ROUTINE EDT$$LNM_CMD (                    ! Execute a line-mode command
;   117          0687  1     COM_START                                     ! Command to execute
;   118          0688  1     ) : NOVALUE =
;   119          0689  1
;   120          0690  1 !++
;   121          0691  1 ! FUNCTIONAL DESCRIPTION:
;   122          0692  1 !
;   123          0693  1 !       This routine executes the command which was last parsed.
;   124          0694  1 !       It is assumed that the command is described in the first
;   125          0695  1 !       node on the parsing stack.
;   126          0696  1 !
;   127          0697  1 ! FORMAL PARAMETERS:
;   128          0698  1 !
;   129          0699  1 !  COM_START            Node block containing the command to execute
;   130          0700  1 !
;   131          0701  1 ! IMPLICIT INPUTS:
;   132          0702  1 !
;   133          0703  1 !       EDT$$G_RCOV_MOD
;   134          0704  1 !       EDT$$Z_PA_STK           The semantic stack containing the command to be executed
;   135          0705  1 !
;   136          0706  1 ! IMPLICIT OUTPUTS:
;   137          0707  1 !
;   138          0708  1 !       EDT$$G_EDIT_MOD
;   139          0709  1 !       EDT$$A_EXE_CURCMD
;   140          0710  1 !       EDT$$Z_EXE_SBLK
;   141          0711  1 !       EDT$$G_EXE_SBITS
;   142          0712  1 !       EDT$$G_EXE_QRYQUIT
;   143          0713  1 !       EDT$$G_EXITD
;   144          0714  1 !       EDT$$G_TXT_ONSCR
;   145          0715  1 !
;   146          0716  1 ! ROUTINE VALUE:
;   147          0717  1 !
;   148          0718  1 !       NONE
;   149          0719  1 !
;   150          0720  1 ! SIDE EFFECTS:
;   151          0721  1 !
;   152          0722  1 !       MANY
;   153          0723  1 !
;   154          0724  1 !--
;   155          0725  1
;   156          0726  2     BEGIN
;   157          0727  2
;   158          0728  2     EXTERNAL ROUTINE
;   159          0729  2         EDT$$SC_NONREVID,                       ! Turn off reverse video
;   160          0730  2         EDT$$STOP_WKINGMSG,                     ! Turn off timer AST
;   161          0731  2         EDT$$CLR_CMD : NOVALUE,                 ! Process the CLEAR command
;   162          0732  2         EDT$$DEFK_CMD : NOVALUE,                ! Process the DEFINE KEY command
;   163          0733  2         EDT$$DEFM_CMD : NOVALUE,                ! Process the DEFINE MACRO command
;   164          0734  2         EDT$$DEL_CMD : NOVALUE,                 ! Process the DELETE command
;   165          0735  2         EDT$$MAC_CALL,                          ! Invoke a macro
;   166          0736  2         EDT$$EXI_CMD : NOVALUE,                 ! Process the EXIT command
;   167          0737  2         EDT$$FILL_CMD : NOVALUE,                ! Process the FILL command
;   168          0738  2         EDT$$FMT_CRLF,                          ! Terminate a line
;   169          0739  2         EDT$$FMT_MSG,                           ! Put the text of a message in the format buffer
;   170          0740  2         EDT$$OUT_HLP,                           ! Get help
```

```
;  171     0741  2          EDT$$INCL_CMD : NOVALUE,                ! Process the INCLUDE command
;  172     0742  2          EDT$$INS_CMD : NOVALUE,                 ! Process the INSERT command
;  173     0743  2          EDT$$DEL_LNS,                           ! Delete a range of lines
;  174     0744  2          EDT$$INS_TXT : NOVALUE,                 ! Insert text
;  175     0745  2          EDT$$MOVCPY_CMD : NOVALUE,              ! Process the MOVE and COPY commands
;  176     0746  2          EDT$$NULL_CMD : NOVALUE,                ! Process the null command
;  177     0747  2          EDT$$PRNT_CMD : NOVALUE,                ! Process the PRINT command
;  178     0748  2          EDT$$RNG_REPOS,
;  179     0749  2          EDT$$RSEQ_CMD : NOVALUE,                ! Process the RESEQUENCE command
;  180     0750  2          EDT$$SET_CMD : NOVALUE,                 ! Process the SET command
;  181     0751  2          EDT$$SHW_CMD : NOVALUE,                 ! Process the SHOW command
;  182     0752  2          EDT$$LN_SUBCMD : NOVALUE,               ! Process the SUBSTITUTE command
;  183     0753  2          EDT$$SUB_NXTCMD : NOVALUE,              ! Process the SUBSTITUTE NEXT command
;  184     0754  2          EDT$$TSADJ_CMD : NOVALUE,               ! Process the TABS ADJUST command
;  185     0755  2          EDT$$TY_CMD : NOVALUE,                  ! Process the TYPE command
;  186     0756  2          EDT$$WR_CMD : NOVALUE;                  ! Process the WRITE command
;  187     0757  2
;  188     0758  2      EXTERNAL
;  189     0759  2          EDT$$G_EDIT_MOD,
;  190     0760  2          EDT$$G_RCOV_MOD,
;  191     0761  2          EDT$$G_TXT_ONSCR,                       ! Use "Press return to continue" after this command is done
;  192     0762  2          EDT$$G_EXITD,
;  193     0763  2          EDT$$A_EXE_CURCMD : REF NODE_BLOCK,     ! Pointer to the current command.
;  194     0764  2          EDT$$G_EXE_QRYQUIT,                     ! Quit flag for /QUERY operations.
;  195     0765  2          EDT$$G_RNG_FRSTLN,
;  196     0766  2          EDT$$A_CUR_BUF : REF TBCB_BLOCK,
;  197     0767  2          EDT$$Z_RNG_ORIGPOS : POS_BLOCK,
;  198     0768  2          EDT$$G_EXE_SBITS,                       ! The options switches.
;  199     0769  2          EDT$$Z_EXE_SBLK : REF NODE_BLOCK;       ! The option switch value block.
;  200     0770  2
;  201     0771  2      MESSAGES ((INVOPTCOM));
;  202     0772  2  !+
;  203     0773  2  ! Define the table containing valid options for each command.
;  204     0774  2  !-
;  205     0775  2
;  206     0776  2      BIND
;  207     0777  2          OPT_TABLE = UPLIT (BYTE (
;  208     0778  2          0,                                      ! Options for null command
;  209     0779  2          0,                                      ! Options for change
;  210     0780  2          OPT_DUPL+OPT_QUERY,                     ! Options for copy
;  211     0781  2          0,                                      ! Options for define
;  212     0782  2          OPT_QUERY,                              ! Options for delete
;  213     0783  2          OPT_SAVE+OPT_SEQ,                       ! Options for exit
;  214     0784  2          0,                                      ! Options for find
;  215     0785  2          0,                                      ! Options for include
;  216     0786  2          0,                                      ! Options for insert
;  217     0787  2          OPT_QUERY,                              ! Options for move
;  218     0788  2          0,                                      ! Options for print
;  219     0789  2          OPT_SAVE,                               ! Options for quit
;  220     0790  2          0,                                      ! Options for replace
;  221     0791  2          OPT_SEQ,                                ! Options for resequence
;  222     0792  2          0,                                      ! Options for set
;  223     0793  2          0,                                      ! Options for show
;  224     0794  2          OPT_QUERY+OPT_NOTYP+
;  225     0795  2              OPT_BRIEF,                          ! Options for SUBSTITUTE
;  226     0796  2          OPT_BRIEF+OPT_STAY,                     ! Options for type
;  227     0797  2          OPT_SEQ,                                ! Options for write
```

```
;   228     0798  2          0,                                  ! Options for subs next
;   229     0799  2          0,                                  ! Options for help
;   230     0800  2          OPT_NOTYP,                          ! Options for verify
;   231     0801  2          0,                                  ! Options for tab adjust
;   232     0802  2          0,                                  ! Options for fill
;   233     0803  2          0,                                  ! Options for def macro
;   234     0804  2          0                                   ! Options for mac call
;   235     0805  2              )): VECTOR [LAST_COM, BYTE];
;   236     0806  2
;   237     0807  2      EDT$$A_EXE_CURCMD = .COM_START;
;   238     0808  2  !+
;   239     0809  2  ! The top level had better be a command code.
;   240     0810  2  !-
;   241     0811  2      ASSERT (.EDT$$A_EXE_CURCMD [NODE_TYPE] EQL COM_NODE);
;   242     0812  2      EDT$$Z_EXE_SBLK = .EDT$$A_EXE_CURCMD [SWITS];
;   243     0813  2
;   244     0814  3      IF (.EDT$$Z_EXE_SBLK NEQ 0)
;   245     0815  2      THEN
;   246     0816  3          BEGIN
;   247     0817  3
;   248     0818  4          IF ((.EDT$$Z_EXE_SBLK [SW_BITS] AND ( NOT .OPT_TABLE [.EDT$$A_EXE_CURCMD [COM_NUM]])) NEQ 0)
;   249     0819  3          THEN
;   250     0820  4              BEGIN
;   251     0821  4              EDT$$FMT_MSG (EDT$_INVOPTCOM);
;   252     0822  4              RETURN;
;   253     0823  3              END;
;   254     0824  3
;   255     0825  3          EDT$$G_EXE_SBITS = .EDT$$Z_EXE_SBLK [SW_BITS];
;   256     0826  3          END
;   257     0827  2      ELSE
;   258     0828  2          EDT$$G_EXE_SBITS = 0;
;   259     0829  2
;   260     0830  2      EDT$$G_EXE_QRYQUIT = 0;
;   261     0831  2
;   262     0832  2      CASE .EDT$$A_EXE_CURCMD [COM_NUM] FROM COM_NULL TO LAST_COM OF
;   263     0833  2          SET
;   264     0834  2
;   265     0835  2          [COM_QUIT] :
;   266     0836  3              BEGIN
;   267     0837  3              EDT$$G_EXITD = 1;
;   268     0838  3              EDT$$FMT_CRLF ();
;   269     0839  2              END;
;   270     0840  2
;   271     0841  2          [COM_INSERT] :
;   272     0842  3              BEGIN
;   273     0843  3              EDT$$INS_CMD ();
;   274     0844  2              END;
;   275     0845  2
;   276     0846  2          [COM_DELETE] :
;   277     0847  3              BEGIN
;   278     0848  3              EDT$$DEL_CMD ();
;   279     0849  2              END;
;   280     0850  2
;   281     0851  2          [COM_TYPE] :
;   282     0852  3              BEGIN
;   283     0853  3              EDT$$STOP_WKINGMSG ();
;   284     0854  3              EDT$$SC_NONREVID ();
```

```
;  285     0855  3              EDT$$TY_CMD ();
;  286     0856  2              END;
;  287     0857  2
;  288     0858  2          [COM_NULL] :
;  289     0859  3              BEGIN
;  290     0860  3              EDT$$STOP_WKINGMSG ();
;  291     0861  3              EDT$$NULL_CMD ();
;  292     0862  2              END;
;  293     0863  2
;  294     0864  2          [COM_FIND] :
;  295     0865  3              BEGIN
;  296     0866  3              EDT$$G_RNG_FRSTLN = 1;
;  297     0867  3              EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS);
;  298     0868  3              EDT$$RNG_REPOS (.EDT$$A_EXE_CURCMD [RANGE1]);
;  299     0869  2              END;
;  300     0870  2
;  301     0871  2          [COM_SUBS] :
;  302     0872  3              BEGIN
;  303     0873  3              EDT$$LN_SUBCMD ();
;  304     0874  2              END;
;  305     0875  2
;  306     0876  2          [COM_EXIT] :
;  307     0877  3              BEGIN
;  308     0878  3              EDT$$EXI_CMD ();
;  309     0879  2              END;
;  310     0880  2
;  311     0881  2          [COM_REPLACE] :
;  312     0882  3              BEGIN
;  313     0883  3
;  314     0884  3              IF EDT$$DEL_LNS () THEN EDT$$INS_TXT ();
;  315     0885  3
;  316     0886  2              END;
;  317     0887  2
;          0888  2          [COM_INCLUDE] :
;  3       0889  3              BEGIN
;  320     0890  3              EDT$$INCL_CMD ();
;  321     0891  2              END;
;  322     0892  2
;  323     0893  2          [COM_WRITE] :
;  324     0894  3              BEGIN
;  325     0895  3              EDT$$G_EXE_SBITS<OPB_STAY> = 1;      ! write never changes the position
;  326     0896  3              EDT$$WR_CMD ();
;  327     0897  2              END;
;  328     0898  2
;  329     0899  2          [COM_SUBS_NEXT] :
;  330     0900  3              BEGIN
;  331     0901  3              EDT$$SUB_NXTCMD ();
;  332     0902  2              END;
;  333     0903  2
;  334     0904  2          [COM_HELP] :
;  335     0905  3              BEGIN
;  336     0906  3              EDT$$SC_NONREVID ();
;  337     0907  3              EDT$$OUT_HLP (.EDT$$A_EXE_CURCMD [FILSPEC], .EDT$$A_EXE_CURCMD [FSPCLEN], 1);
;  338     0908  3              EDT$$G_TXT_ONSCR = 1;
;  339     0909  2              END;
;  340     0910  2
;  341     0911  2          [COM_SET] :
```

```
;  342     0912  3              BEGIN
;  343     0913  3              EDT$$SET_CMD ();
;  344     0914  3              END;
;  345     0915  2
;  346     0916  2          [COM_SHOW] :
;  347     0917  3              BEGIN
;  348     0918  3              EDT$$SHW_CMD ();
;  349     0919  3              END;
;  350     0920  2
;  351     0921  2          [COM_CHANGE] :
;  352     0922  3              BEGIN
;  353     0923  3      !+
;  354     0924  3      ! Position to the first line of the specified range
;  355     0925  3      ! and set the current mode to change mode.
;  356     0926  3      !-
;  357     0927  3              EDT$$G_RNG_FRSTLN = 1;
;  358     0928  3              EDT$$CPY_MEM (POS_SIZE, .EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS);
;  359     0929  3
;  360     0930  3              IF EDT$$RNG_REPOS (.EDT$$A_EXE_CURCMD [RANGE1]) THEN EDT$$G_EDIT_MOD = CHANGE_MODE;
;  361     0931  3
;  362     0932  3              END;
;  363     0933  2
;  364     0934  2          [COM_MOVE] :
;  365     0935  3              BEGIN
;  366     0936  3              EDT$$MOVCPY_CMD (1);
;  367     0937  3              END;
;  368     0938  2
;  369     0939  2          [COM_COPY] :
;  370     0940  3              BEGIN
;  371     0941  3              EDT$$MOVCPY_CMD (0);
;  372     0942  3              END;
;  373     0943  2
;  374     0944  2          [COM_DEFINE] :
;  375     0945  3              BEGIN
;  376     0946  3              EDT$$DEFK_CMD ();
;  377     0947  3              END;
;  378     0948  2
;  379     0949  2          [COM_PRINT] :
;  380     0950  3              BEGIN
;  381     0951  3              EDT$$G_EXE_SBITS<OPB_STAY> = 1;      ! print never changes the position
;  382     0952  3              EDT$$PRNT_CMD ();
;  383     0953  3              END;
;  384     0954  2
;  385     0955  2          [COM_RESEQ] :
;  386     0956  3              BEGIN
;  387     0957  3              EDT$$RSEQ_CMD ();
;  388     0958  3              END;
;  389     0959  2
;  390     0960  2          [COM_TADJ] :
;  391     0961  3              BEGIN
;  392     0962  3              EDT$$TSADJ_CMD ();
;  393     0963  3              END;
;  394     0964  2
;  395     0965  2          [COM_FILL] :
;  396     0966  3              BEGIN
;  397     0967  3              EDT$$FILL_CMD ();
;  398     0968  2              END;
```

```
;  399          0969  2
;  400          0970  2          [COM_DEF_MAC] :
;  401          0971  3              BEGIN
;  402          0972  3              EDT$$DEFM_CMD ();
;  403          0973  2              END;
;  404          0974  2
;  405          0975  2          [COM_MAC_CALL] :
;  406          0976  3              BEGIN
;  407          0977  3              EDT$$MAC_CALL (.EDT$$A_EXE_CURCMD [RANGE1]);
;  408          0978  2              END;
;  409          0979  2
;  410          0980  2          [COM_CLEAR] :
;  411          0981  3              BEGIN
;  412          0982  3              EDT$$CLR_CMD ();
;  413          0983  2              END;
;  414          0984  2
;  415          0985  2          [OUTRANGE] :
;  416          0986  3              BEGIN
;  417          0987  3              ASSERT (0);
;  418          0988  2              END;
;  419          0989  2          TES;
;  420          0990  2
;  421          0991  2      IF .EDT$$G_EXITD
;  422          0992  2      THEN
;  423          0993  2
;  424          0994  2          IF .EDT$$G_EXE_SBITS<OPB_SAVE> THEN EDT$$G_EXITD = 3;
;  425          0995  2
;  426          0996  2      IF .EDT$$G_RCOV_MOD THEN EDT$$G_EXITD = 0;
;  427          0997  2
;  428          0998  1      END;                                        ! of routine EDT$$LNM_CMD

                                                        .TITLE  EDT$LXCOM EDT$LXCOM - execute a line-mode comma
;                                                                       nd
                                                        .IDENT  \V04-000\

                                                        .PSECT  _EDT$CODE,NOWRT,  SHR,  PIC,2

00  10  00  40  00  02  00  00  00  50  02  00  22  00  00  00000 P.AAA:  .BYTE   0, 0, 34, 0, 2, 80, 0, 0, 0, 2, 0, 64, 0, -
                 00  00  00  00  08  00  00  10  84  0E  00  0000F                16, 0, 0, 14, -124, 16, 0, 0, 8, 0, 0, 0, -
                                                                            0

                                                OPT_TABLE=          P.AAA
                                                        .EXTRN  EDT$$SC_NONREVID
                                                        .EXTRN  EDT$$STOP_WKINGMSG
                                                        .EXTRN  EDT$$CLR_CMD, EDT$$DEFK_CMD
                                                        .EXTRN  EDT$$DEFM_CMD, EDT$$DEL_CMD
                                                        .EXTRN  EDT$$MAC_CALL, EDT$$EXI_CMD
                                                        .EXTRN  EDT$$FILE_CMD, EDT$$FMT_CRLF
                                                        .EXTRN  EDT$$FMT_MSG, EDT$$OUT_ALP
                                                        .EXTRN  EDT$$INCL_CMD, EDT$$INS_CMD
                                                        .EXTRN  EDT$$DEL_LNS, EDT$$INS_TXT
                                                        .EXTRN  EDT$$MOVCPY_CMD
                                                        .EXTRN  EDT$$NULL_CMD, EDT$$PRNT_CMD
                                                        .EXTRN  EDT$$RNG_REPOS, EDT$$RSEQ_CMD
                                                        .EXTRN  EDT$$SET_CMD, EDT$$SHW_CMD
                                                        .EXTRN  EDT$$LN_SUBCMD, EDT$$SOB_NXTCMD
```

```
                                                        .EXTRN  EDT$$TSADJ_CMD, EDT$$TY_CMD
                                                        .EXTRN  EDT$$WR_CMD, EDT$$G_EDIT_MOD
                                                        .EXTRN  EDT$$G_RCOV_MOD
                                                        .EXTRN  EDT$$G_TXT_ONSCR
                                                        .EXTRN  EDT$$G_EXITD, EDT$$A_EXE_CURCMD
                                                        .EXTRN  EDT$$G_EXE_QRYQUIT
                                                        .EXTRN  EDT$$G_RNG_FRSTLN
                                                        .EXTRN  EDT$$A_CUR_BUF, EDT$$Z_RNG_ORIGPOS
                                                        .EXTRN  EDT$$G_EXE_SBITS
                                                        .EXTRN  EDT$$Z_EXE_SBLK
                                                        .EXTRN  EDT$_INVOPTCOM, EDT$$INTER_ERR

                                    0FFC 00000          .ENTRY  EDT$$LNM_CMD, Save R2,R3,R4,R5,R6,R7,R8,R9,-; 0686
                                                                R10,R11
                5B 00000000G   00    9E 00002          MOVAB   EDT$$INTER_ERR, R11                           ;
                5A 00000000G   00    9E 00009          MOVAB   EDT$$Z_EXE_SBLK, R10                          ;
                59 00000000G   00    9E 00010          MOVAB   EDT$$G_EXITD, R9                              ;
                58 00000000G   00    9E 00017          MOVAB   EDT$$A_EXE_CURCMD, R8                         ;
                57 00000000G   00    9E 0001E          MOVAB   EDT$$G_EXE_SBITS, R7                          ;
                68         04    AC  D0 00025          MOVL    COM_START, EDT$$A_EXE_CURCMD                  ; 0807
                50               68  D0 00029          MOVL    EDT$$A_EXE_CURCMD, R0                         ; 0811
                01               60  91 0002C          CMPB    (R0), #1                                      ;
                                 03  13 0002F          BEQL    1$                                            ;
                6B               00  FB 00031          CALLS   #0, EDT$$INTER_ERR                            ;
                50               68  D0 00034 1$:      MOVL    EDT$$A_EXE_CURCMD, R0                         ; 0812
                6A           14  A0  D0 00037          MOVL    20(R0), EDT$$Z_EXE_SBLK                       ;
                51               6A  D0 0003B          MOVL    EDT$$Z_EXE_SBLK, R1                           ; 0814
                                 29  13 0003E          BEQL    3$                                            ;
                50           01  A0  9A 00040          MOVZBL  1(R0), R0                                     ; 0818
                50           9E AF40  9A 00044          MOVZBL  OPT_TABLE[R0], R0                             ;
                50               50  D2 00049          MCOML   R0, R0                                        ;
                50           04  A1  D3 0004C          BITL    4(R1), R0                                     ;
                                 0E  13 00050          BEQL    2$                                            ;
                          00000000G  8F  DD 00052          PUSHL   #EDT$_INVOPTCOM                           ; 0821
                00000000G  00        01  FB 00058          CALLS   #1, EDT$$FMT_MSG                          ;
                                     04  0005F          RET                                                   ; 0820
                50               6A  D0 00060 2$:      MOVL    EDT$$Z_EXE_SBLK, R0                           ; 0825
                67           04  A0  D0 00063          MOVL    4(R0), EDT$$G_EXE_SBITS                       ;
                                 02  11 00067          BRB     4$                                            ; 0814
                                 67  D4 00069 3$:      CLRL    EDT$$G_EXE_SBITS                              ; 0828
                          00000000G  00  D4 0006B 4$:      CLRL    EDT$$G_EXE_QRYQUIT                        ; 0830
                56               68  D0 00071          MOVL    EDT$$A_EXE_CURCMD, R6                         ; 0832
             19              00    01  A6  8F 00074          CASEB   1(R6), #0, #25                            ;
0150         0145            0116       006E     00079 5$:      .WORD   10$-5$,-                              ;
00C5         007E            00A9       004E     00081                  26$-5$,-                               ;
0039         0159            0141       0045     00089                  30$-5$,-                               ;
010D         0104            0166       00B2     00091                  32$-5$,-                               ;
00DB         00CE            0057       00A0     00099                  8$-5$,-                                ;
0178         016F            0196       00E4     000A1                  13$-5$,-                               ;
                             018A       0181     000A9                  11$-5$,-                               ;
                                                                        17$-5$,-                               ;
                                                                        7$-5$,-                                ;
                                                                        29$-5$,-                               ;
                                                                        34$-5$,-                               ;
                                                                        6$-5$,-                                ;
                                                                        15$-5$,-                               ;
                                                                        35$-5$,-                               ;
```

```
                                                
```

```
                                                7B  11 00184          BRB     40$                                     ; 0832
                                00000000G  00        00  FB 00186 24$:  CALLS   #0, EDT$$SHW_CMD                        ; 0918
                                                7E  11 0018D 25$:     BRB     42$                                     ; 0832
                                00000000G  00        01  D0 0018F 26$:  MOVL    #1, EDT$$G_RNG_FRSTLN                   ; 0927
                                           50 00000000G  00  D0 00196          MOVL    EDT$$A_CUR_BUF, R0                      ; 0928
                     00000000G  00           60        0E  28 0019D          MOVC3   #14, (R0), EDT$$Z_RNG_ORIGPOS           ;
                                                04  A6  DD 001A5          PUSHL   4(R6)                                   ; 0930
                                00000000G  00        01  FB 001A8          CALLS   #1, EDT$$RNG_REPOS                      ;
                                           64        50  E9 001AF 27$:     BLBC    R0, 44$                                 ;
                                              00000000G  00  D4 001B2          CLRL    EDT$$G_EDIT_MOD                         ;
                                                5C  11 001B8 28$:     BRB     44$                                     ; 0832
                                                01  DD 001BA 29$:     PUSHL   #1                                      ; 0936
                                                02  11 001BC          BRB     31$                                     ;
                                                7E  D4 001BE 30$:     CLRL    -(SP)                                   ; 0941
                                00000000G  00        01  FB 001C0 31$:  CALLS   #1, EDT$$MOVCPY_CMD                    ;
                                                4D  11 001C7          BRB     44$                                     ; 0832
                                00000000G  00        00  FB 001C9 32$:  CALLS   #0, EDT$$DEFK_CMD                      ; 0946
                                                44  11 001D0 33$:     BRB     44$                                     ; 0832
                                           67        80  8F 88 001D2 34$:  BISB2   #128, EDT$$G_EXE_SBITS                ; 0951
                                00000000G  00        00  FB 001D6          CALLS   #0, EDT$$PRNT_CMD                       ; 0952
                                                37  11 001DD          BRB     44$                                     ; 0832
                                00000000G  00        00  FB 001DF 35$:  CALLS   #0, EDT$$RSEQ_CMD                       ; 0957
                                                2E  11 001E6          BRB     44$                                     ; 0832
                                00000000G  00        00  FB 001E8 36$:  CALLS   #0, EDT$$TSADJ_CMD                      ; 0962
                                                25  11 001EF          BRB     44$                                     ; 0832
                                00000000G  00        00  FB 001F1 37$:  CALLS   #0, EDT$$FILL_CMD                       ; 0967
                                                1C  11 001F8 38$:     BRB     44$                                     ; 0832
                                00000000G  00        00  FB 001FA 39$:  CALLS   #0, EDT$$DEFM_CMD                       ; 0972
                                                13  11 00201 40$:     BRB     44$                                     ; 0832
                                                04  A6  DD 00203 41$:  PUSHL   4(R6)                                   ; 0977
                                00000000G  00        01  FB 00206          CALLS   #1, EDT$$MAC_CALL                       ;
                                                07  11 0020D 42$:     BRB     44$                                     ; 0832
                                00000000G  00        00  FB 0020F 43$:  CALLS   #0, EDT$$CLR_CMD                        ; 0982
                                           07        69  E9 00216 44$:  BLBC    EDT$$G_EXITD, 45$                       ; 0991
                          03               67        06  E1 00219          BBC     #6, EDT$$G_EXE_SBITS, 45$               ; 0994
                                           69        03  D0 0021D          MOVL    #3, EDT$$G_EXITD                        ;
                                        02 00000000G  00  E9 00220 45$:     BLBC    EDT$$G_RCOV_MOD, 46$                    ; 0996
                                                69  D4 00227          CLRL    EDT$$G_EXITD                            ;
                                                    04 00229 46$:     RET                                             ; 0998

; Routine Size:  554 bytes,    Routine Base:  _EDT$CODE + 001A

;  429          0999  1
;  430          1000  1 !<BLF/PAGE>
```

```
;  432          1001  1 END                                   ! of module EDT$LXCOM
;  433          1002  1
;  434          1003  0 ELUDOM



                                                    EDT$$LNM_CMD_NOOVERLAY_REF==
                                                                EDT$$LNM_CMD

;                               PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  _EDT$CODE                         580  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)


;                              Library Statistics
;                                        -------- Symbols --------      Pages       Processing
;       File                             Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[EDT.SRC]EDT.L32;1           377       122        32          40       00:00.2
;  _$255$DUA28:[EDT.SRC]PSECTS.L32;1          2         1        50           7       00:00.1



;                                    COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:LXCOM/OBJ=OBJ$:LXCOM MSRC$:LXCOM.BLI/UPDATE=(ENH$:LXCOM)

; Size:          554 code + 26 data bytes
; Run Time:          00:25.1
; Elapsed Time:      00:29.9
; Lines/CPU Min:      2396
; Lexemes/CPU-Min:    8454
; Memory Used:   160 pages
; Compilation Complete
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LXCOMNOO LIS

MCGETLIN LIS

MCRIGHT LIS

MCCHANGE LIS

MACCAL LIS

NOOPEN LIS

PRAPPNUM LIS

MSGTXT LIS

PRGETTOK LIS

PRISTOK LIS

LXPRINT LIS

MCTOP LIS

MCDOWN LIS

MCLEFT LIS

MCBOTTOM LIS

PAUDIT LIS

PRGETCHR LIS

LXCOM LIS

MAIN LIS

MCUP LIS